REMARKS

Claims 1 through 36 are pending and have been examined. The Specification was objected to because pages 2, 4, 6, 8, 10, and 12 are missing. Claims 1, 3 through 13, 15 through 25, and 27 through 36 were rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent 5,627,970 ("Keshav"). Claims 2, 14, and 26 were rejected under 25 U.S.C. 103(a) as being obvious over Keshav in view of U.S. Patent 5,526,350 ("Gittins").

The Applicants have amended claims 1, 13, and 25 to better distinguish the prior art by incorporating the limitations of former claims 2, 14, and 26, and claims 2, 14, and 26 have been amended to round out the scope of protection claimed by the Applicants. Support for these amendments may be found, e.g., at pages 6 through 9 of the Specification, where an embodiment concerning MPEG compressed video is disclosed. No new matter has been added.

In view of these amendments, and the remarks set forth below, the Applicants respectfully request that the Examiner reconsider his rejection of claims 1 through 36.

I. Objection to the Specification

In paragraph 1 of the Office Action, the Examiner objects to the Specification as missing even-numbered pages. In response to the Examiner's objection, the Applicants submit, along with this paper, even-numbered pages 2, 4, 6, 8, 10, and 12 of the Specification. These pages were originally filed with this Application and no new matter has been added. In view of this submission, Applicants respectfully request withdrawal of the objection to the Specification.

II. Rejections Under Section 102

In paragraphs 2 and 3 of the Office Action, claims 1, 3 through 13, 15 through 25, and 27 through 36 were rejected as anticipated by Keshav. The Applicants respectfully request reconsideration of this rejection.

Of the claims rejected in paragraphs 2 and 3, claims 1, 13 and and 25 are independent claims. As amended, independent claim 1 is directed to a method for transmitting data in real time from a sender to a receiver in a digital communications network, comprising the steps of maintaining an estimate of bandwidth available from the sender to the receiver; and adjusting transmission based on the estimate in order to maintain real time transmission. Likewise, amended claims 13 and 25 are directed to systems for transmitting data in real time from a sender to a receiver in a digital communications network.

With these arrangements, as described for example on pages 1 and 2 of the Specification, a transmission technique is provided which, although not perfectly reliable, makes it more likely that transmitted data arrive at its destination on time. Such a technique is highly preferred for congestion control in a digital communications network such as the Internet or corporate "Intranets."

No such transmission technique is disclosed or suggested by Keshav. Keshav describes an invention that attempts to be 100% reliable, meaning that any packet sent from the source node will, eventually, be received at the destination node. This is not a real time protocol, since if a packet is lost by the network (which occurs quite often), Keshav will retransmit the packet to ensure 100% reliability. In a real time environment, that packet is only valuable if it arrives on time. For example, in video conferencing a packet containing speech data must arrive at the same time as the packet containing video data of the person uttering the speech. Since Keshav specifically discloses the automatic retransmission of lost packets, e.g., at Col. 6, ll. 37 – 45, Keshav incurs increasing delays that are intolerabloe for real time environments. Nothing in Keshav discloses or suggests the real time transmission techniques required by amended claims 1, 13, or 25.

Claims 3 through 12, 15 through 24, and 27 through 36 each depends from, and thus include all of the limitations of, either independent claim 1, 13 or 25. Thus Kesav also fails to anticipate these claims as well. Moreover, while claims 10, 12, 22, 24, 24 and 36 each recite retransmitting a packet, retransmission of packets can occur only if such retransmission does not damage the real time service. This is in contrast to Keshav, which discloses without exception retransmitting data that has been lost in the network.

III. Rejections Under Section 103

In paragraphs 4 and 5 of the Office Action, claims 2, 14 and 26 were rejected as being obvious over Keshav in view of Gittins. The Applicants respectfully request reconsideration of these rejections as well.

Claims 2, 14, and 26 depend from independent claims 1, 13 and 25 respectively. Thus, as discussed above, Keshav is deficient with respect to claims 2, 14, and 26 because it does not disclose or suggest the limitation of real time transmission as required by these claims.

Gittens does not cure this deficiency in Keshav. Rather, Gittens describes a technique which requires operation under an entirely different environment and is not comparable with the present invention. Gittens is directed to "[a] switched telecommunications network [which] includes a plurality of switches for switching different types of traffic . . ." which is not Internet (i.e., TCP/IP – based) traffic. Moreover, Gittins assumes that bandwidth is managed by the bandwidth managers centrally, and that the bandwidth managers assign bandwidth to each connection. In contrast, the present invention does not presuppose any of this preexisting infrastructure of bandwidth managers, since neither the Internet nor Intranets have the infrastructure of bandwidth managers, and instead must "maintain an estimate of bandwidth available" in the communications network.

Thus, nothing in either Keshav or Gittins discloses or suggests the real time transmission techniques required by claims 2, 14 and 26. As result, the Examiner has not made out a prima facie case of obviousness with respect to these claims. The Applicants respectfully request that the rejection of claims 2, 14, and 26 under Section 103 be withdrawn.

IV. Conclusion

For the reasons set forth above, applicant respectfully submits that this application is now in condition for allowance. Reconsideration and prompt allowance are respectfully requested.

Respectfully submitted,

BAKER BOTTS L.L.P.

By: ______________________

Paul A. Ragusa
Patent Office Reg. No. 38,587
Attorney for Applicants
212-408-2588

Encl.

a transmission technique is preferred which is not perfectly reliable, but which makes it more likely that the data arrive on time. The technique uses an estimate of the bandwidth which is available in a network, from a sender to a receiver. The estimate is increased or decreased, by the sender, depending on monitoring of acknowledgments from the receiver.

The technique is compatible with TCP, and its use by a sender in a connection results in fair sharing of network resources with all other connections. It can be used, e.g., with well-established protocols such as File Transfer Protocol (FTP) and Hyper Text Transfer Protocol (HTTP).

Brief Description of the Drawing

Fig. 1 is a representation of packet format for a preferred embodiment of the invention.

Fig. 2 is a flow chart for processing at a network server, in accordance with a preferred embodiment of the invention.

Figs. 3a and 3b are schematics of communications systems in accordance with preferred embodiments of the invention, with fixed and adaptable bandwidth requirements, respectively.

Fig. 4 is a flow chart for exemplary rate control processing in a system according to Fig. 3b.

Fig. 5a is a graphic representation of system behavior for an example in a system in accordance with Fig. 3a.

Fig. 5b is a graphic representation of system behavior for an example in a system in accordance with Fig. 3b.

the protocol, thus decreasing the number of outstanding packets.

Optionally, selective retransmission can be provided for. A current estimate is maintained of the round trip time, i.e. the time elapsed between sending a packet and receiving an acknowledgment. The protocol sends the estimate to the receiver in each packet header. When the receiver determines that a packet has been lost, it then determines if there is enough time to receive the retransmitted packet before it is needed. If so, the receiver can request a retransmission; otherwise, no request is made. In real-time audio or video, for example, if the receiver has 100 milliseconds worth of data buffered for playback when detecting loss of a packet, and if the estimate for the round-trip time is less than 100 milliseconds, a request for retransmission is likely to result in timely retransmission of the lost packet. Thus, a best-effort attempt is made at reliability.

As illustrated by Fig. 1, a data packet includes the standard User Datagram Protocol (UDP) header, a 2-byte sequence number, a 4-byte time stamp, and a 4-byte round-trip time estimate measured in milliseconds. The sequence number is for packet reordering at the receiver, in case packets arrive out of order. The time stamp is media dependent and generally provides an indication of the presentation time of the packet.

Fig. 2 illustrates preferred packet processing by a server system. There is a main loop which continually checks whether (i) data can be sent out, (ii) an acknowledgment has arrived, (iii) a timeout has occurred, or (iv) a retransmission was requested. Initially, CWND is set to the size of the first packet to be transmitted, ensuring that the first packet can be sent out.

later, in Congestion Avoidance Phase (Phase≠SS), CWND is increased by the square of the size divided by the current value of CWND:

$$CWND = CWND + size^2/CWND.$$

Slow Start calls for increasing the value of CWND each time an acknowledgment is received. In the case of variable length packets, with CWND being the number of bytes of outstanding packets, Slow Start calls for increasing the value of CWND by the size of the packet to which the acknowledgment refers.

After increasing CWND, there follows checking of CWND > SSThresh, the Slow Start Threshold. If true, Phase = CA, for Congestion Avoidance.

Then, concerning timeouts, if an acknowledgment is not received within Timeout (TO) milliseconds after it was sent, the packet is determined to be lost in the network and the appropriate action is taken. This includes (i) setting SSThresh to half of the current CWND, (ii) setting CWND to the value of the next packet to be sent out (i.e. resetting CWND), (iii) setting Phase to Slow Start, (iv) decreasing the outstanding acknowledgment by the size of the packet which timed out, and (v) doubling the Timeout period (TO).

Finally, the system checks for receipt of a retransmission request. If so, it resends the appropriate data and resets SSThresh and CWND to half the current value of CWND. This is known as Fast Recovery. The system then returns to check for further data to send, and whether CWND > ACK.

As described, the technique does not depend on whether the bandwidth requirements of the media can be changed or adapted. Fig. 3a shows a system with non-adaptable media, such as MPEG. The server reads the media from a file or obtains it from a live source and

For MPEG video, another technique for bandwidth reduction is known as Dynamic Rate Shaping (DRS) as described by A. Eleftheriadis et al., "Constrained and General Dynamic Rate Shaping of Compressed Digital Video", Proceedings, 2nd IEEE International Conference on Image Processing, Washington, D.C., October 1995, pp. III.396-399. This involves identifying, frame by frame, those coefficients in the MPEG stream which are least important in terms of image quality, and removing them from the stream.

Fig. 3b shows an adaptable media system. Again, the original media either is stored locally or is supplied by a live source. But, in this case the data enters a media adaptation module which shapes the media into an estimate of the available bandwidth. The shaped media enters the buffer, which is then read by the media pump. Again, the media pump sends out data so as to comply with the CWND. At the client, the data is buffered for presentation to the user. The client provides feedback information for congestion control.

The status of the buffer between the media adaptation module and the media pump is critical for this system. If the buffer is filling, then the media pump is sending data out more slowly than the media adaptation module is filling the buffer. In this case, the system should decrease the bandwidth requirements of the media so that the buffer does not overflow, by dropping frames or assigning a lower rate to DRS.

Conversely, if the buffer is emptying, the media pump is sending data out faster than the buffer is being filled by the media adaptation module. In this case, the system should increase the bandwidth requirements of the media so that the user gets the best quality possible. Since rate control provides information to the media adaptation module, it is highly dependent on the time of

of the average occupancy. The coefficient of variation is defined as

$$\text{Variance(samples)}/\text{Mean}^2\text{(samples)},$$

where the samples are the two values of the average buffer occupancy. Beta is then multiplied by 10. If Beta is less than 0.1, it is assigned the value 0.1, if it is greater that 1.0, it is assigned the value 1.0.

Finally, the new transmission rate is calculated by subtracting, from the previous rate, the value Beta·Centering·Diff·8, where the factor 8 is due to Diff being in bytes and the rate being in bits. These steps are repeated every 5 seconds.

Adaptable media can cope with more drastic variations in network resources, as compared with non-adaptable media. In non-adaptable media, a decrease in network resources results in less data reaching the receiver than is needed, and the receiver can rely only on its initial buffering to continue playback.

Fig. 5a shows an example of using a non-adaptive media. In this case, the rate of the media is 300 kbps, and the final buffering is 5 seconds (1500 kb). The available bandwidth is continually changing. In the beginning there is just enough bandwidth for the media and no buffering is used. But as soon as the available bandwidth decreases to 200 kbps, the receiver must begin using its buffering. If the bandwidth stays low for an extended period of time, the buffer may become completely depleted, at which time the user will experience an interruption in playback. This occurs at around 40 seconds. The available bandwidth then increases to 350 kbps, at which time the buffer can accumulate again.

With adaptable media, the initial buffering has to be used only when the bandwidth requirements of the media cannot be reduced further. As illustrated by Fig. 5b, for the same rate and initial buffering as in Fig. 5a,

the receiver when a packet has arrived containing an error. Internet Protocol (IP) packets are simply dropped at the receiver if there is an error in the header.

Currently, with UDP, the receiver system has the option of instructing the sender system not to put error checking in packets. This is on a system-wide basis, so that all UDP packets coming from the sender system will not use error checking, which is undesirable when other applications expect UDP error checking.

Preferably, in accordance with a preferred embodiment of the invention, the receiver can distinguish whether a packet is lost due to congestion or error, in an application-specific fashion.

Fig. 6 illustrates a packet constructed from an IP packet provided with the shaded area by the operating system. Error checking will be over the IP header only, so that a bit error there still results in the packet being dropped without notification. However, without error checking over the payload, a bit error in the payload does not result in the packet being dropped.

In this embodiment of the invention, the sender constructs a UDP header inside the payload of the IP packet, for the packet to appear as a regular UDP packet at the receiver. In the UDP header, the sender sets the Cyclic Redundancy Code (CRC) field to zero, indicating that no error checking is used. Accordingly, when the receiver reads the packet, the UDP module of the receiver system will not do any error checking, leaving it to the application to check for errors.

So that packets received with errors are not used, the sender must insert its own error checking functionality into the payload of the UDP packet it constructs. In Fig. 6, this is shown as Application Defined CRC. If, using Application Defined CRC, the receiver determines that there is an error, the receiver